Requested Patent: GB2319151A

Title:

SOCKET BINDING METHOD IN COMMUNICATION SYSTEM USING SOCKET FUNCTION ;

Abstracted Patent: US5941950 ;

Publication Date: 1999-08-24 ;

Inventor(s):

KIM JONG-HOON (KR); HWANG YEONG-GIL (KR); KIM SOO-HYUN (KR) ;

Applicant(s): SAMSUNG ELECTRONICS CO LTD (KR) ;

Application Number: US19970954618 19971020 ;

Priority Number(s): KR19960052637 19961107 ;

IPC Classification: G06F11/00 ;

Equivalents:

CN1099183B, CN1182240, DE19748009, FR2755560, JP10187567, RU2171491 ;

ABSTRACT:

A computer network system includes a network program for establishing an inter-system communication between clients and the server by using a socket function. If a 'bind ( )' system call results in failure in the course of making the inter-system communication by using the socket function, the 'bind ( )' system call is retried after a lapse of a predetermined time. The predetermined time is a time determined based on a time delay in closing the abnormal socket.

(12) UK Patent Application (19) GB (11) 2 319 151 (13) A

(43) Date of A Publication 13.05.1998

(21) Application No 9723131.0

(22) Date of Filing 03.11.1997

(30) Priority Data
(31) 96052637 (32) 07.11.1996 (33) KR

(51) INT CL[6]
H04L 29/06

(52) UK CL (Edition P)
H4P PPEC
U1S S2124

(56) Documents Cited
GB 2259387 A GB 2232855 A EP 0613274 A2
WO 95/08903 A1

(58) Field of Search
UK CL (Edition P) H4P PPEC PPND
INT CL[6] H04L 29/06 , H04Q 11/04
Online:WPI, CLAIMS, USPATFULL

(71) Applicant(s)
Samsung Electronics Co Limited

(Incorporated in the Republic of Korea)

416 Maetan-dong, Paldal-gu, Suwon-city, Kyonggi-do, Republic of Korea

(72) Inventor(s)
Yeong-Gill Hwang
Soo-Hyun Kim
Jong-Hoon Kim

(74) Agent and/or Address for Service
Dibb Lupton Alsop
Fountain Precinct, Balm Green, SHEFFIELD, S1 1RZ, United Kingdom

(54) Retrying network socket after predetermined time

(57) A method and apparatus for assigning a socket in a network system is described. According to the method, if an attempt to assign a socket fails, another attempt is not made until a pre-determined time has elapsed.

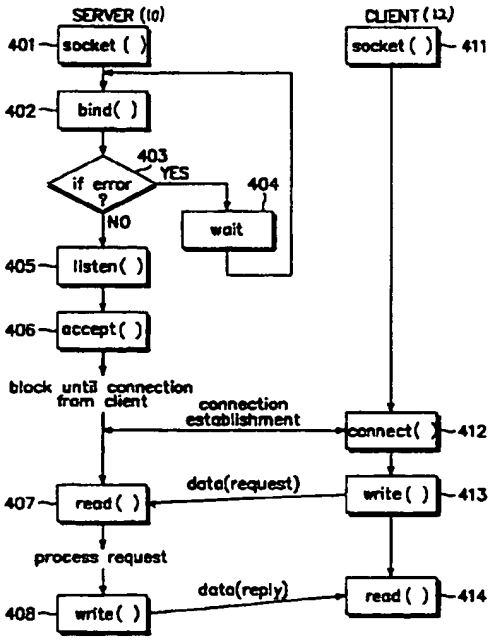

FIG. 4

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

This print takes account of replacement documents submitted after the date of filing to enable the application to comply with the formal requirements of the Patents Rules 1995

GB 2 319 151 A

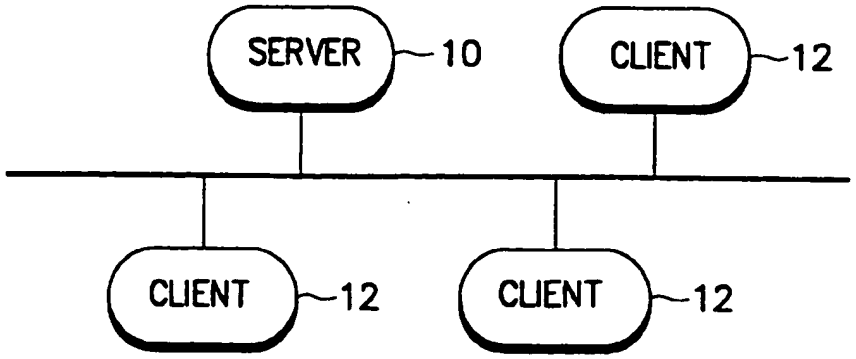
SERVER
10
CLIENT
12
CLIENT
12
CLIENT
12

FIG. 1

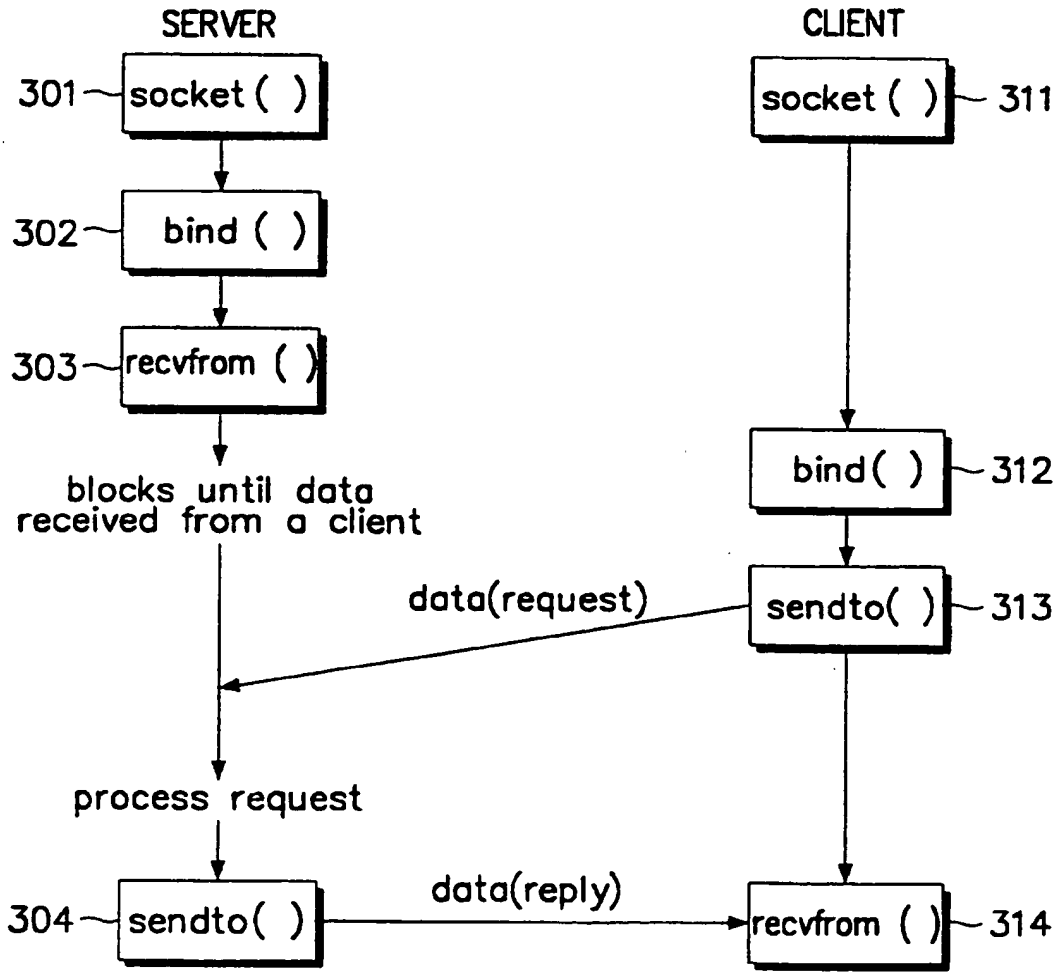
SERVER
CLIENT
301
socket ( )
socket ( )
311
302
bind ( )
303
recvfrom ( )
bind( )
312
blocks until data
received from a client
data(request)
sendto( )
313
process request
304
sendto( )
data(reply)
recvfrom ( )
314

FIG. 3

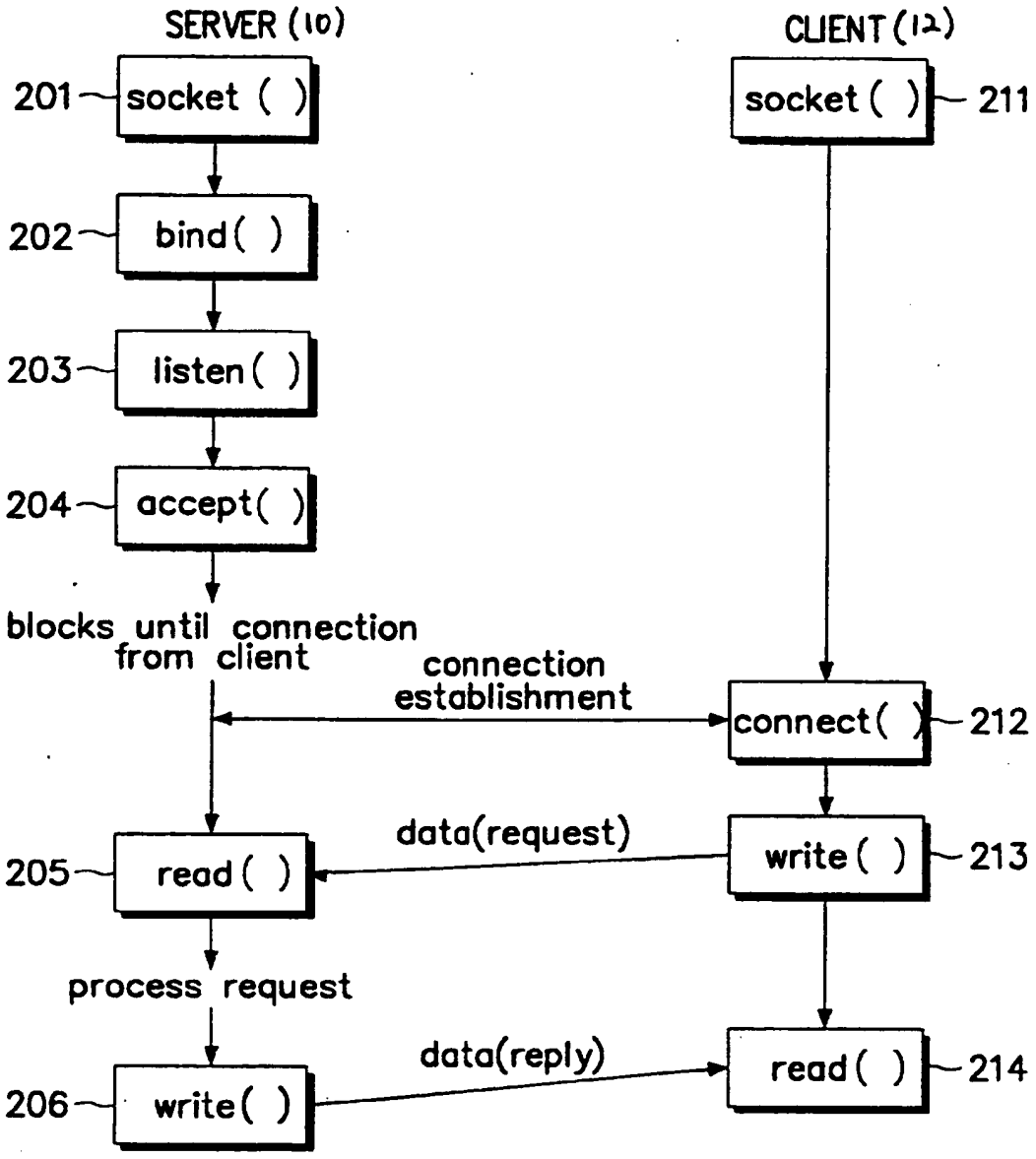
SERVER (10)
CLIENT (12)
201
socket ( )
socket ( )
211
202
bind( )
203
listen( )
204
accept( )
blocks until connection from client
connection establishment
connect( )
212
data(request)
205
read( )
write( )
213
process request
data(reply)
206
write( )
read( )
214

FIG. 2

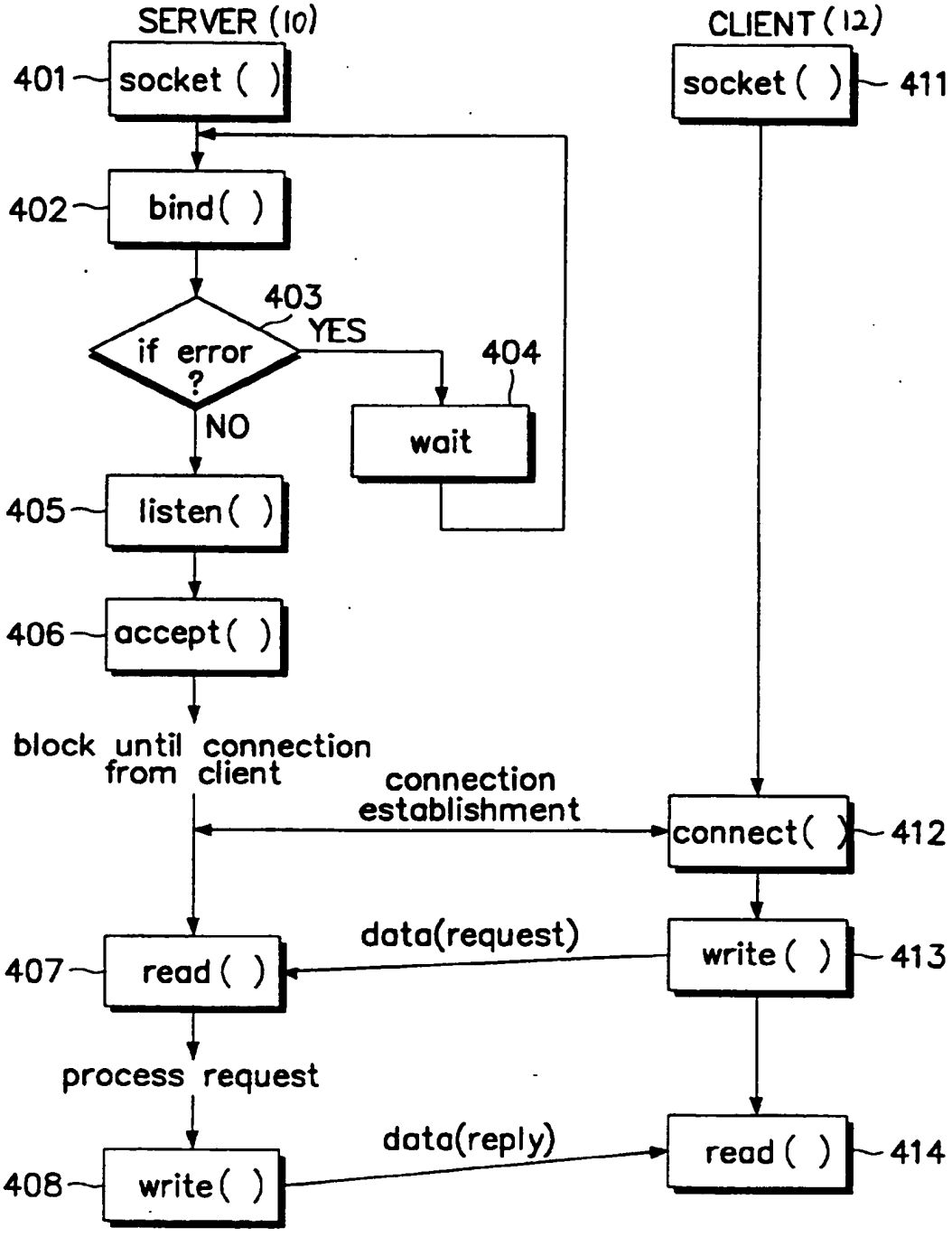
SERVER (10)
CLIENT (12)
401
socket ( )
socket ( )
411
402
bind( )
403
YES
if error ?
404
wait
NO
405
listen( )
406
accept( )
block until connection from client
connection establishment
connect( )
412
data(request)
write ( )
413
407
read ( )
process request
data(reply)
read ( )
414
408
write( )

FIG. 4

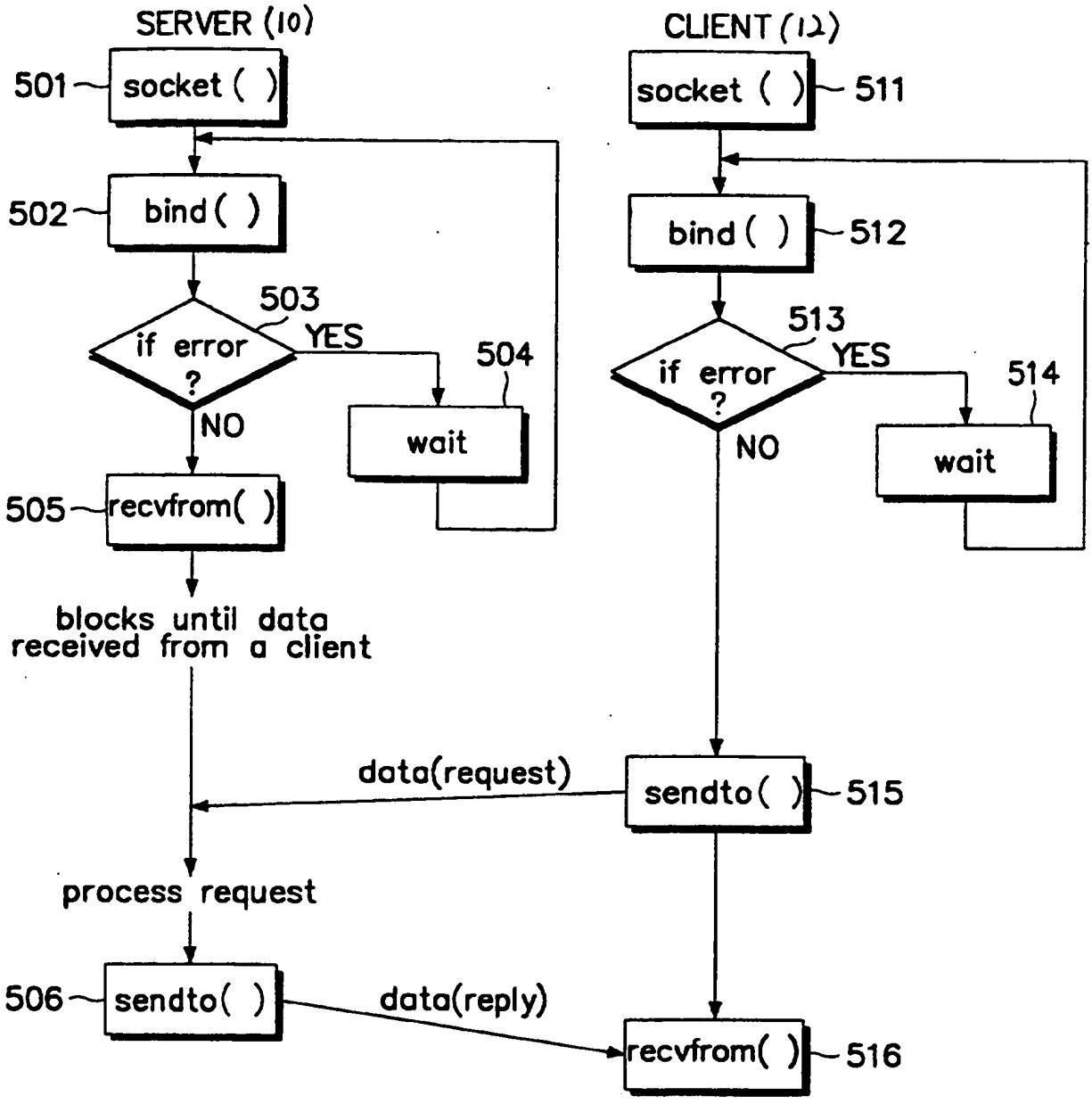
SERVER (10)
CLIENT (12)
501
socket ( )
socket ( )
511
502
bind ( )
bind ( )
512
503
if error ?
YES
NO
504
wait
513
if error ?
YES
NO
514
wait
505
recvfrom( )
blocks until data received from a client
data(request)
sendto( )
515
process request
506
sendto( )
data(reply)
recvfrom( )
516

FIG. 5

SOCKET ASSIGNMENT METHOD

Field of the Invention

The present invention relates to inter-system communication in computer networks, and more particularly to a method of establishing communication by using the socket function of a network program.

Background to the Invention

In order to establish inter-system communication in a computer network, a network program is generally used. The network program establishes communication by using a socket function which assigns sockets to each communication task. For example, a 'bind ()' system call is commonly used for assigning the socket.

Fig. 1 shows a computer network system which uses the 'bind ()' system call. The computer network system includes a server 10 and a plurality of clients 12 connected to the server 10. The server 10 is an independent computer system for processing job requests from clients 12. The clients 12 are also independent computer systems which request the server 10 to process a particular job. The inter-system communication between server 10 and the clients 12 is managed by a network program contained on both the server 10 and the clients 12. In order to establish communication between the server 10 and any one of the clients 12, a socket function in the network program is used, and a socket is assigned to the specific communication task.

A prior art method for assigning a socket is illustrated in Figs. 2 and 3. Figs. 2 and 3 illustrate flow charts representing the process of assigning a socket by using a 'bind ()' system call (her inafter, referred to as socket binding). Fig. 2 shows the method for a connection-oriented protocol system, and Fig. 3 shows the method for a connectionless protocol system.

As stated above, Fig. 2 shows the process f socket binding in a connection-oriented protocol system. The server 10 and th client 12 specify a communication protocol type at steps 201 and 211, respectively [socket ()]. The server 10 assigns a unique name to an unnamed socket at step 202 [bind ()]. The server 10 notifies the client 12 that it is ready to accept a connection from the client 12, at step 203 [listen ()], and then waits to accept a substantial connection from the client 12, at step 204 [accept ()]. Client 12 establishes a connection to the server 10 by using a socket descriptor at step 212 [connect ()], and writes request data by using the socket descriptor designated by the socket (), at step 213 [write ()]. Server 10 reads the request data by using the socket descriptor designated by the socket () at step 205 [read ()], and thereafter executes the process according to the data request. Once executed, server 10 writes reply data according to the process execution at step 206 [write ()]. Client 12 then reads the reply data from server 10, at step 214 [read ()].

Fig. 3 shows the process of socket binding in a connectionless protocol system. The server 10 and client 12 specify a communication protocol type at steps 301 and 311, respectively [socket ()]. The server 10 and the client 12 assign a unique name to an unnamed socket at steps 302 and 312, respectively [bind ()]. The client 12 then sends request data by using the socket descriptor designated by the socket (), at step 313 [sendto ()]. Server 10 receives the request data from client 12 by using the socket descriptor designated by the socket (), at step 303 [recvfrom ()], and upon receipt, server 10 executes the process according to the request. Thereafter, the server 10 sends the reply data corresponding to the data request, at step 304 [sendto ()]. Client 12 receives the reply data from the s rver 10 at step 314 [recvfrom ()].

If the 'bind ()' system call results in failure when assigning a unique name to an unnamed socket, the 'bind ()'

system call will be retried immediately. However, if the 'bind ()' system call is repeatedly re-attempted and the attempts repeatedly fail, a time d lay is caused and a system call error will be generated during the time delay. The prior art method repeatedly retries the system call even in the case of system abnormalities. This retrying unnecessarily wastes system resources as well as increasing the load on the system.

Summary of the Invention

It is therefore an object of the present invention to provide a method which reduces the number of repeat system calls when establishing inter-system communication by using a socket function.

It is another object of the present invention to provide a method which avoids the waste of system resources when establishing inter-system communication by using a socket function.

It is still another object of the present invention to provide a method which prevents a load increase on the system when establishing an inter-system communication by using a socket function.

Accordingly, the present invention provides a method of assigning a socket in a network system using a socket function, the method comprising the steps of:

activating a pre-determined time delay in response to a failed socket assignment attempt; and

retrying the socket assignment after the pre-determined time delay has elapsed.

Preferably, the duration of the time delay is based on the time it takes the system to close an abnormal socket.

The network system may comprises a connection- riented protocol system. Alternatively, it may comprise a connectionless protocol system.

Socket assignment may comprises assigning a unique name to an unnamed socket.

The present invention also extends to a network apparatus including a socket function comprising:

means for activating a pre-determined time delay in response to a failed socket assignment attempt; and

means for retrying the socket assignment after the pre-determined time delay has elapsed.

Preferably, the duration of the time delay is based on the time it takes the apparatus to close an abnormal socket.

The apparatus may operate with a connection-oriented protocol or it may operate with a connectionless protocol.

Socket assignment may comprise assigning a unique name to an unnamed socket.

Brief Description of the Drawings

The present invention will now be described by way of example with reference to the accompanying drawings in which:

Fig. 1 is a block diagram showing a general computer network system;

Fig. 2 is a flow chart of a socket binding method for a connection-oriented protocol system according to the prior art;

Fig. 3 is a flow chart of a socket binding method for a connectionless protocol system according to the prior art;

Fig. 4 is a flow chart of a socket binding method for a connection-oriented protocol system according to an illustrative embodiment of the present invention; and

Fig. 5 is a flow chart of a socket binding method for a connectionless protocol according to another illustrative embodiment of the present invention.

Detailed Description of Preferred Embodiments

A socket binding method according to the invention is shown in Fig.s 4 and 5. Fig. 4 illustrates a flow chart of the socket binding process for a connection-oriented protocol system. Fig. 5 illustrates a flow chart for the socket binding process for a connectionless protocol system. The socket binding method is characterized in that when the 'bind ()' system call results in failure, the 'bind ()' system call is not immediately retried, but is retried after a predetermined time has elapsed. The predetermined time is based on the time it takes the system to close an abnormal socket.

As shown in Fig. 4, the server 10 and the client 12 specify a communication protocol type at steps 401 and 411, respectively [socket ()]. The server 10 assigns a unique name to an unnamed socket at step 402 [bind ()], and checks, at step 403, whether the 'bind ()' system call has resulted in success or failure. If the 'bind ()' system call is successful, the server 10 goes to step 405. On the other hand, if the 'bind ()' system call fails, server 10 goes to step 404 to wait for a predetermined time and thereafter returns to step 402 to retry the 'bind ()' system call. The predetermined time refers to the time delay of the system itself, i.e. the time required to close the initially abnormal socket.

The server 10 notifies the client 12 at step 405 that it is ready to accept a connection from client 12 [listen ()], and then waits to accept a substantial connection from the client 12, at step 406 [accept ()]. Client 12 establishes a connection to the server 10 by using a socket descriptor at step 412 [connect ()], and writes request data by using the socket descriptor designated by the socket (), at step 413 [write ()]. Then, server 10 reads the request data by using the socket descriptor designated by the socket () at step 407 [read ()], and xecut s the process according to the data request. Thereafter, server 10 writes reply data according to the executed process at step 408 [write ()].

Client 12 reads the reply data from the server 10, at step 414 [read ()].

Fig. 5 shows the process of socket binding for a connectionless protocol system. The server 10 and client 12 specify a communication protocol type at steps 501 and 511, respectively [socket ()]. The server 10 and the client 12 assign a unique name to an unnamed socket at steps 502 and 512, respectively [bind ()]. The server 10 and the client 12 then check whether the 'bind ()' system call has resulted in success or failure, at steps 503 and 513, respectively. If the 'bind ()' system call is successful, server 10 and client 12 go to steps 505 and 515, respectively. Otherwise, if the 'bind ()' system call fails, server 10 and client 12 go to steps 504 and 514 respectively, to wait for a predetermined time and thereafter return to the steps 502 and 512 respectively, to retry the 'bind ()' system call. Here, the predetermined time refers to the time delay of the system itself, as mentioned above.

If the 'bind ()' system call is successful at step 513, client 12 sends request data by using the socket descriptor designated by the socket (), at step 515 [sendto ()]. Further, if the 'bind ()' system call is successful at step 503, server 10 receives the request data by using the socket descriptor designated by the socket (), at step 505 [recvfrom ()]. Upon receiving the request data from client 12, server 10 executes the process according to the data request. Thereafter, server 10 sends the reply data corresponding to the data request, at step 506 [sendto ()]. Then, client 12 receives the reply data from server 10 at step 516 [recvfrom ()].

As described above, if the 'bind ()' system call attempt fails, the 'bind ()' system call is re-attempted only after a predetermined time has elapsed. Therefore, the present invention avoids unnecessary waste of system resources as well as avoiding unnecessary increases in system load.

Claims:

1. A method of assigning a socket in a network system using a socket function, the method comprising the steps of:

activating a pre-determined time delay in response to a failed socket assignment attempt; and

retrying the socket assignment after the pre-determined time delay has elapsed.

2. A method according to claim 1, where the duration of the time delay is based on the time it takes the system to close an abnormal socket.

3. A method according to claim 1 or claim 2, in which the network system comprises a connection-oriented protocol system.

4. A method according to claim 1 or claim 2, in which the network system comprises a connectionless protocol system.

5. A method according to any preceding claim in which socket assignment comprises assigning a unique name to an unnamed socket.

6. A network apparatus including a socket function comprising:

means for activating a pre-determined time delay in response to a failed socket assignment attempt; and

means for retrying the socket assignment after the pre-determined time delay has elapsed.

7. An apparatus according to claim 6 in which the duration of the time delay is based on the time it takes the apparatus to close an abnormal socket.

8. An apparatus according to claim 6 or claim 7 which operates with a connection-oriented protocol.

9. An apparatus according to claim 6 or claim 7 which operates with a connectionless protocol.

10. An apparatus according to any preceding claim in which socket assignment comprises assigning a unique name to an unnamed socket.

11. A method substantially as described herein with reference to Fig. 4 or Fig. 5.

12. An apparatus including means for performing a method substantially as described herein with reference to Fig. 4 or Fig. 5.

The Patent Office

9

Application No:	GB 9723131.0	**Examiner:**	B.J.SPEAR
Claims searched:	1-12	**Date of search:**	4 February 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): H4P (PPEC, PPND)

Int Cl (Ed.6): H04L 29/06, H04Q 11/04

Other: Online:WPI, CLAIMS, USPATFULL

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
Y	GB2259387A	(Motorola) Whole document, eg claim 1 lines 6-11.	1-10
Y	GB2232855A	(Sec. Of State for Defence).Whole document, eg claim 1	1-10
Y	EP0613274A2	(IBM)Whole document, eg Figs. 1 and 6.	1-10
Y	WO95/08903A1	(Codex) Whole document, eg claim 2 lines 12-13, Fig. 10.	1-10

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.